מטלת מנחה 12 - מבני נתונים ומבוא לאלגוריתמים

שאלה 1

נציע אלגוריתם למיון מערך עם שגיאה בגודל k כאשר עומד לרשותנו זיכרון חיצוני בגודל $O(k)$.
השימוש בזיכרון החיצוני נדרש משום שלא ניתן להשתמש בk התאים הראשונים במערך, שכן כפי שנראה בהמשך, אילו היינו משתמשים בהם ערכם היה "נדרס" על ידי ערכי k התאים האחרונים במערך. האלגוריתם משתמש בשגרות $Max\_Heap\_Insert(A,\ key)$ ו $Heap\_Extract\_Max(A)$, שיעילות זמן הריצה של שתיהן היא $O(\lg n)$ כאשר $n = A.size$.

הרעיון המרכזי של האלגוריתם: יצירת "תור קדימויות" בגודל k של איברים, לפי גודלם.

לפי ההנחה על המערך, ידוע לנו שהאיבר המקסימלי במערך נמצא באחד מ$k + 1$ התאים האחרונים במערך. נבנה ערימה בגודל $k + 1$ מתאים אלו, נוציא את האיבר המקסימלי ונשבצו במקום האחרון. (נוכל "לדרוס" את הערך הקיים שם לפני בבטחה, שכן הכנסנו אותו לערמה קודם לכן והוא שמור שם).

נחזור על הפעולה - נכניס ל"תור" את האיבר במקום ה $n - k - 1$, נוציא את האיבר הכי גדול בתור וכך הלאה. בסוף, כשלא נותרו איברים להכניס, נשבץ את $k + 1$ האיברים שנותרו בתור ב $k + 1$ התאים האחרונים, בסדר הנכון, וסיימנו.

כך, בכל איטרציה של הלולאה המרכזית, ניתן לחלק את המערך ל-3 חלקים:

האלגוריתם:

$Heap\_Sort\_Error(A,\ k)$

$\triangleright$ 1. $Create\ queue\ of\ the\ k\ largest\ elements\ in\ A$

$Heap \leftarrow create\ heap[k + 1]$

$for\ i\ \leftarrow A.length\ to\ (A.length - k - 1)$

$\quad Max\_Heap\_Insert(Heap,\ A[i])\ \triangleright Enqueue\ element$

$\triangleright$ 2. $Iterate\ over\ all\ cells\ from\ n\ to\ k\ and\ populate\ value\ from\ the\ queue$

$for\ i\ \leftarrow A.length\ to\ k + 1$

$\quad A[i] = Heap\_Extract\_Max(Heap) \triangleright Dequeue\ largest\ element\ in\ queue$

$\quad Max\_Heap\_Insert(Heap,\ A[i - k - 1]) \triangleright Enqueue\ next\ element\ instead$

$\triangleright$ 3. $Populate\ the\ rest\ of\ the\ queue\ members,\ largest\ to\ smallest$

$for\ i \leftarrow k + 1\ to\ 1\ inclusive$

$\quad A[i] = Heap\_Extract\_Max(Heap) \triangleright Dequeue\ largest\ element\ in\ queue$

הוכחת נכונות:

* הערה - נסמן ב $A[i]$ את האיבר במקום ה i במערך בזמן הווה, וב $A'[i]$ את הערך שהיה בו לפני תחילת המיון.

1. בשלב הראשון, אנחנו בונים תור המכיל את הערכים שהיו בתאים $[n - k, n]$ במערך. ברור שצעד זה נכון, משום שאנחנו עוברים איבר איבר ומוסיפים אותו לתור. לפי נכונות השגרה Max_Heap_Insert, ברור גם שהתור מסודר מהאיבר הגדול לקטן.
2. בשלב השני, שלב הלולאה המרכזית, ננסח את השמורה הבאה (נשים לב שהאינדקס רץ מהסוף להתחלה).

<u>**שמורה:**</u> בסוף האיטרציה הi,

(i) תת-המערך $A[i,.., n]$ ממוין כהלכה (כלומר: $A[i] \leq ... \leq A[n]$),

(ii) כל איבר q הנמצא בתור מקיים $q \leq A[i]$

(iii) תת-המערך $A[1,..., i - k - 1]$ עדיין מקיים את הנתון

נוכיח את השמורה באינדוקציה.

<u>אתחול (בסיס האינדוקציה):</u> לאחר האיטרציה הn, הערך הגדול ביותר שהיה ב"תור" הוצא ממנו הוצב במקום ה $A[n]$, ולאחר מכן הוכנס ל"תור" האיבר $A[n - k]$.

(i) ברור שמתקיים $A[n] \leq A[n]$ כנדרש.

(ii) האיברים שהיו בתור לפני תחילת האיטרציה קטנים בוודאות מהאיבר שהוצא ממנו - כי הוא האיבר המקסימלי בתור הקדימויות. ובפרט - $A'[n] \leq A[n]$. לפי הנתון על המערך עבור $n, n - k - 1$, מתקיים תמיד $n - (n - k - 1) = k + 1 > k$ ולכן $A[n - k - 1] \leq A'[n] \leq A[n]$ ובפרט, לפי טרנזיטיביות, $A[n - k - 1]$ הנוסף לתור קטן מ-$A[n]$ כנדרש.

(iii) מאחר ולא החלפנו אף איבר בטווח $A[1,..., n - k - 1]$, תת-מערך זה עדיין מקיים את הנתון.

<u>צעד האינדוקציה:</u> נניח את נכונות השמורה באיטרציה ה$n \geq i + 1 \geq k + 2$ ונוכיח את נכונותה עבור $k + 1 \leq i \leq n - 1$.

במהלך האיטרציה הוצאנו את האיבר המקסימלי בתור ושיבצנו אותו במקום ה $A[i]$ במערך.

(i) מהנחת האינדוקציה, $A[i]$ שהיה שייך לתור בסוף האיטרציה הקודמת מקיים $A[i] \leq A[i + 1]$. כמו כן לפי הנחת האינדוקציה $A[i + 1] \leq ... \leq A[n]$ ולכן לפי טרנזיטיביות $A[i] \leq ... \leq A[n]$ כנדרש.

(ii) האיברים שהיו בתור לפני תחילת האיטרציה קטנים בוודאות מהאיבר המקסימלי בתור הקדימויות. לפי טענת העזר, קיים בוודאות $i \leq m$ כך ש$A'[m] \leq A[i]$ ולכן לפי הנתון על המערך עבור $(i - k - 1), m$ מתקיים

$k + 1 > k = (- k - 1) - i \geq (i - k - 1) - m \Leftarrow A[i - k - 1] \leq A'[m]$

ולכן לפי טרנזיטיביות $A[i - k - 1] \leq A[i]$ כנדרש.

<u>סיום:</u> נכונות השמורה מעידה על כך שבאיטרציה ה$k + 1$:

(i) תת-המערך $A[k + 2,..., n]$ ממוין כהלכה.

(ii) שאר $k + 1$ איברי המערך, שכעת נמצאים בתור, קטנים כולם מ $A[k + 2]$.

3. בשלב השלישי, כאשר נוציא את $k + 1$ האיברים שנותרו בתור, נקבל לפי נכונות שגרת הוצאת האיבר המקסימלי מהתור ש $A[k + 1] \leq ... \leq A[1]$. לפי נכונות השמורה, $A[k + 2] \leq ... \leq A[n]$ וכן $A[k + 1] \leq A[k + 2]$ ולכן לפי טרנזיטיביות $A[1] \leq ... \leq A[n]$ והמערך ממוין כהלכה.

<u>הוכחת טענת העזר:</u> ניעזר בהנחת האינדוקציה - המערך $A[i+1,...,n]$ ממוין כהלכה וכל ערך הנמצא בתור לפני תחילת האיטרציה הi, ובפרט הערך ב $A[i]$ לאחר האיטרציה, קטן מכל איברי תת-המערך.
נניח בשלילה כי כל הערכים שהיו ב $A'[i,...,n]$ גדולים מ$A[i]$. מדובר ב $n-i+1$ ערכים, מתוכם $n-i$ לכל היותר מאוכלסים ב$A[i+1,...,n]$. כלומר, קיים איבר אחד לפחות, שמיקומו המקורי היה באינדקס i ומעלה (ולכן בוודאות ערך זה נכנס לתור), ואינו אוכלס ב $A[i+1,...,n]$ (ולכן עדיין בתור לפני האיטרציה הi).
מאחר וערך זה גדול ממש מ$A[i]$, הפעלת השגרה $Heap\_Extract\_Max$ על התור המכיל שני ערכים אלה, והעובדה שהשגרה החזירה את הערך הקטן מביניהם, סותרת את נכונותה!
אבל הוכח שהשגרה נכונה, לכן הנחת השלילה שגויה.

סיבוכיות זמן-ריצת האלגוריתם:

1. בניית תור בגודל $k+1$, לפי פרק 6 בספר, היא פעולה שסיבוכיות זמן הריצה שלה $O(k+1)=O(k)$.
2. גוף הלולאה מתבצע $(n-k-1)$ פעמים.
   2.1. הוצאת האיבר המקסימלי מהתור היא פעולה שיעילות זמן הריצה שלה במקרה הגרוע היא $O(\lg(k+1))$ - לפי הספר.
   2.2. הכנסת איבר חדש לתור היא פעולה שסיבוכיות זמן הריצה שלה גם היא $O(\lg(k+1))$ במקרה הגרוע - לפי הספר.
3. גוף הלולאה מתבצע $k+1$ פעמים.
   3.1. הוצאת האיבר המקסימלי התור, תיקח גם כאן $O(\lg(k+1))$ במקרה הגרוע ביותר.

סך הכל:

$$T(n)=k+(n-k-1)\cdot\lg(k+1)\ +\ (k+1)\cdot\lg(k+1)=$$
$$=k+n\lg(k+1)\ \leq_{(1)} n+n\lg(k+1) =_{(2)} O(n\lg(k+1)) =_{(3)} O(n\lg k)$$

הסברי מעברים:

1. ברור ש $k\leq n$, לא ייתכן שהבדל אינדקסי שני איברים שאינם "בסדר הנכון" יהיה יותר מגודל המערך.
2. מתקיים כנדרש $(\lg(k+1)+1)\cdot n\leq c\cdot n\lg(k+1)$ עבור $\lg k+1\leq c\cdot\lg(k+1)\Rightarrow 1+\frac{1}{\lg(k+1)}\leq c$ כאשר $c=2$, משום ש $\lg(k+1)\geq\lg 2=1$ ולכן $\frac{1}{\lg(k+1)}\leq 1$.
3. ניתן לזנוח את הקבוע 1.

הדבר מתיישב עם העובדה שמיון בעזרת ערמה של מערך ממוין עם שגיאה בגודל n (כלומר: מערך לא ממוין), הוא בעל סיבוכיות זמן ריצה של $O(n\lg n)$.

שאלה 2

האלגוריתם לבניית הערימה

נבנה ערימת מינימום כך ששורת העלים שלה היא המערך, כאמור בהדרכה, ולכל קומה מעליה יש חצי מכמות הצמתים בקומה מתחתיה. נניח שמספר האיברים במערך, ובהתאם גם מספר העלים, הוא חזקה של 2.

גודל הערימה - $1 + 2 + 4 + ... + n = \sum_{i=0}^{\lg n} 2^i = \frac{1(2^{\lg n+1}-1)}{2-1} = 2n - 1$ לפי סכום סדרה הנדסית. לכן נדרש זיכרון חיצוני בסדר גודל של $\Theta(n)$ כנדרש.

בערימה, כל צומת שאינה עלה תכיל את ערכו של הבן המינימלי שלה. כך ניצור וריאציה של ערימת מינימום.

$Build\_Query\_Heap(A)$

$Query\_Heap \leftarrow create\ array[2 \cdot A.length - 1]$

$\triangleright\ Copy\ original\ array\ members\ to\ leaves.\ Leaf\ indexes\ start\ at\ \lceil \frac{2n-1}{2} \rceil = n$

$for\ i \leftarrow 1\ to\ A.size$

$Query\_Heap[i + A.size - 1] = A[i]$

$\triangleright\ Compute\ the\ rest\ of\ the\ nodes$

$for\ i \leftarrow 1\ to\ A.size - 1$

$Query\_Heap[i] = minimum(Query\_Heap[Left(i)],\ Query\_Heap[Right(i)])$

$return\ Query\_Heap.$

סיבוכיות זמן-ריצה

הפעולה	סיבוכיות זמן-ריצה
יצירת מערך הערימה	$O(1)$
העתקת כל איברי המערך לעלים של מערך הערימה	$3n = O(n)$
חישוב שאר איברי הערימה	$6n = O(n)$

סך הכל - סיבוכיות זמן הריצה היא $O(n)$ כנדרש.

מענה של שאילתות

כעת, בהינתן הערימה, ניתן לענות על שאילתות מהסוג $min(i, j)$ בסיבוכיות זמן ריצה של $O(\lg n)$. נוכיח זאת.

הסבר על האלגוריתם:
ניעזר בערימה שיצרנו ו "נטפס" בה לפי בצורך. על מנת לטפס ולא לפגוע בנכונות האלגוריתם, יש לוודא עבור שני צמתי הקיצון שהוריהם יכסו טווח המוכל בטווח הרצוי*. משוידאנו את זה, נוכל להשוות את הוריהם של הצמתים וכך לחסוך מספר רב מאוד של השוואות!. נחזור על הפעולה גם עם הוריהם של הצמתים עד שנצמצם את טווח ההשוואה.

* למשל, עבור הטווח $[3, 7]$, ההורה של צומת מספר 7 במערך המקורי, יהיה $min(A[7], A[8])$. אילו $A[8] < A[7]$, אז אם נשווה את ההורה של $A[7]$ נקבל את הטווח $[3, 8]$, ודבר זה עלול לפגוע בנכונות האלגוריתם. לכן, נשווה את $A[7]$ עם המינימום הנוכחי, ונשנה את הטווח להיות $[3, 6]$. כאשר נשווה את המינימום בטווח זה עם $A[7]$, נקבל את האיבר המינימלי בטווח $[3, 7]$ כרצוי.

האלגוריתם:

$Min\_Query(A,\ i,\ j,\ Query\_Heap)$

$\quad \triangleright\ Compute\ the\ actual\ indexes\ in\ the\ tree$
$\quad low \leftarrow i\ +\ A.size$
$\quad high \leftarrow j\ +\ A.size$
$\quad min \leftarrow \infty\ \triangleright\ So\ that\ any\ value\ would\ be\ smaller\ than\ it\ in\ initial\ comparison$

$\quad \triangleright\ Find\ minimal\ values\ as\ long\ as\ there\ is\ a\ range\ between\ low\ and\ high$
$\quad while\ low\ \leq\ high\ do$

$\quad\quad \triangleright\ \ If\ low\ points\ to\ a\ right\ child,\ its\ parent\ might\ hold\ a\ value\ that\ is\ not\ in\ the\ range.$
$\quad\quad \triangleright\ Compare\ it\ to\ the\ minimum\ and\ increase\ low\ to\ point\ to\ a\ left\ child$
$\quad\quad if\ low\ =\ Right(Parent(low))\ then$
$\quad\quad\quad min \leftarrow minimum(min,\ Query\_Heap[low])$
$\quad\quad\quad low \leftarrow low\ \ +\ 1$

$\quad\quad \triangleright\ \ If\ high\ points\ to\ a\ left\ child,\ its\ parent\ might\ hold\ a\ value\ that\ is\ not\ in\ the\ range.$
$\quad\quad \triangleright\ Compare\ it\ to\ the\ minimum\ and\ decrease\ high\ to\ point\ to\ a\ right\ child$
$\quad\quad if\ high\ =\ Left(Parent(low))\ then$
$\quad\quad\quad min \leftarrow minimum(min,\ Query\_Heap[high])$
$\quad\quad\quad high \leftarrow high\ \ -\ 1$

$\quad\quad \triangleright\ Now\ it's\ ensured\ that\ low\ and\ high\ cover\ only\ nodes\ with\ parent\ in\ the\ range\ as\ well$
$\quad\quad low \leftarrow Parent(low)$
$\quad\quad high \leftarrow Parent(high)$

$\quad return\ min$

הוכחת נכונות האלגוריתם

זוהי אינה הוכחה פורמלית, אך עם זאת נשתמש ברעיון האינדוקציה כדי לעזור להסביר מדוע האלגוריתם נכון. נרוץ באינדוקציה על k החל מ$k = 0$, ונראה שהאלגוריתם מבצע מספר השוואות מספק הנדרש כדי לקבוע את האיבר המינימלי בטווח $[i, i + k]$ כלשהו.

בסיס האינדוקציה: בטווח $[i, i]$, העלה שהאינדקס שלו במערך המקורי הוא i הוא בן ימני או שמאלי, ולכן בכל מקרה יושווה לאינסוף ויימצא קטן ממנו, וביציאה מהלולאה יתקיים $min = A[i]$ כנדרש.

שלב האינדוקציה: נניח כי האלגוריתם מבצע מספר השוואות מספק כדי לקבוע את האיבר המינימלי בטווח $[i, i + k]$. נוכיח עבור $[i, i + k + 1]$.

אילו $A[i + k + 1]$ בן שמאלי בערימת העזר, אז הוא יושווה עם המינימום הנוכחי באיטרציה הראשונה של הלולאה, ונוסף על כך, לפי הנחת האינדוקציה, יבוצעו מספיק השוואות עם שאר האיברים על מנת לקבוע את האיבר המינימלי.

אילו $A[i + k + 1]$ בן שמאלי בערימת העזר ולכן קיימת צומת שערכה $minimum(A[i + k], A[i + k + 1])$. אילו צומת זו היא בן שמאלי בערימת העזר, אז היא תושווה ובהתאם למקרה הקודם יבצעו מספיק השוואות.

אילו היא בן ימני, נחזור על התהליך עד שנגיע לבן שמאלי והוא יושווה, ואז לפי הנחת האינדוקציה יתקיימו מספיק השוואות על מנת לקבוע את האיבר המינימלי בטווח. אילו לא הגענו לבן שמאלי, אזי הגענו לראש ערימת העזר, כאשר ערך צומת זה הוא הערך המינימלי במערך כולו ולכן יהיה מינימלי בטווח.

סיבוכיות זמן-ריצה

מחוץ ללולאה, אנחנו מבצעים עבודות שסיבוכיות זמן הריצה שלהן היא $O(1)$.
בגוף הלולאה, אנחנו מבצעים עבודות שסיבוכיות זמן הריצה שלהן היא $O(1)$.
הלולאה מתבצעת לכל היותר מספר הפעמים שניתן לחלק את i ואת j ב-2, ובמקרה הגרוע כאשר $j = n$, הלולאה תתבצע לכל היותר $O(\lg n)$ פעמים.

לכן,
$$T(n) = 1 \cdot \lg n + 1 = O(\lg n)$$
כנדרש בדרישות היעילות.

שאלה 3

א. טענה: בשגרת החלוקה $Partition$, שני איברים a, b הקטנים או שווים לאיבר הציר, ישמרו על הסדר היחסי שלהם לאחר החלוקה.

הסבר: יהיו שני איברים a, b הקטנים או שווים לאיבר הציר.

בכל איטרציה של הלולאה בשגרה, לפני האינדקס הנוכחי i (כולל) יימצאו כל האיברים שעברו עליהם עד כה במערך והם קטנים או שווים לאיבר הציר.

כאשר הלולאה תעבור על a הקטן או שווה לאיבר הציר, היא תשבץ אותו באינדקס i_1 כלשהו - האינדקס אותו מחזיק המשתנה i בעת ההחלפה.

<u>לאחר מכן,</u> הלולאה תעבור על b הקטן או שווה גם הוא לאיבר הציר.

מאחר ואיבר זה קטן מאיבר הציר, תתבצע ההוראה $i \leftarrow i + 1$ ולכן לאחר ביצועה בוודאות $i > i_1$. כעת, תתבצע הוראת ההחלפה והאיבר b יימצא בתא שהאינדקס שלו הוא i.

כאמור, $i > i_1$ והסדר היחסי נשמר.

ב. טענה: בשגרת החלוקה $Partition$, שני איברים a, b הגדולים מאיבר הציר, לא בהכרח ישמרו על הסדר היחסי שלהם לאחר החלוקה.

דוגמה: נתבונן במערך $[4, 5, 3]$. בעת הפעלת שגרת ה $Partition$, מאחר ושני האיברים הראשונים במערך גדולים מאיבר הציר 3 ולכן לא תתבצע שום החלפה בעת ביצוע הלולאה.

לאחר הלולאה, האיבר 4 יוחלף עם האיבר 3 ולכן מערך הפלט הוא $[3, 5, 4]$.

היות ושני האיברים $4, 5$ גדולים מאיבר הציר והסדר היחסי שלהם לא נשמר (בפלט 4 לפני 5 ואילו בקלט 5 לפני 4), אי אפשר להניח שהסדר היחסי של שני איברים הגדולים מאיבר הציר יישמר לאחר הפעלת שגרת ה $Partition$.

ג. קלט: המערך $[m + 1, m + 2, ... n, 1, 2, ... m]$ כאשר $m > \frac{n}{2}$.

סיבוכיות זמן ריצה: לאחר ביצוע ה $Partition$ הראשון, שסיבוכיות זמן הריצה שלו היא $O(n)$, עבור איבר החצי m, כל האיברים הקטנים מ-m ישמרו על הסדר היחסי שלהם ותת-המערך של האיברים הקטנים מאיבר החצי ייראה כך: $[1, 2, ..., m - 1]$.

כעת, ייקראו שתי הקריאות הרקורסיביות לשני תתי-המערכים. נסמן ב$T_{>}(n)$ את סיבוכיות זמן הריצה של הקריאה עבור תת-המערך של האיברים הגדולים מ-m, וב $T_{<}(n)$ את סיבוכיות זמן-הריצה עבור תת-המערך של האיברים הקטנים מ-m.

נוסחת סיבוכיות זמן-הריצה: $T(n) = T_{>}(n - m - 1) + T_{<}(m - 1) + O(n)$

<u>נחשב חסם תחתון עבור סיבוכיות זמן הריצה</u>

תת-המערך של האיברים הקטנים מ-m הוא תת-מערך ממוין בסדר עולה. כל קריאה רקורסיבית עליו תגרור בחירה של איבר החצי להיות האיבר האחרון (שהוא האיבר המקסימלי), מעבר על כל האיברים חוץ ממנו, והחלפתם עם עצמם. נוסחת הנסיגה של הקריאות הרקורסיביות:

$$T_{<}(n) = T(n - 1) + O(n) \Rightarrow T_{<}(n) = \sum_{k=1}^{n} k = \frac{n(n+1)}{2} = O(n^2)$$

ולכן, $T(m - 1) = O((m - 1)^2)$.

נחשב חסם תחתון לנוסחה זו המובע באמצעות n. מהנתון $m > \frac{n}{2}$

$$T_{<}(m - 1) = (m - 1)^2 = m^2 - 2m + 1 = O(m^2) > O(\frac{n^2}{4}) = O(n^2)$$

לכן, סיבוכיות זמן הריצה עבור חצי המערך השמאלי חסומה מלמטה על ידי $O(n^2)$.

אם נוסיף "עוד עבודה" של ה $Partition$ הראשוני והעבודה על חצי המערך הימני, נקבל שסך כל סיבוכיות זמן הריצה חסומה מלמטה על ידי $O(n^2)$.

<u>נוסף על החסם התחתון</u>, סיבוכיות זמן הריצה של מיון מהיר, לכל קלט, חסומה מלמעלה על ידי $O(n^2)$.
לכן מתקיים $O(n^2) \leq T(n) \leq O(n^2)$ ולכן בוודאות $T(n) = O(n^2)$ עבור הקלט הנתון.

שאלה 4

א. ננסה לחסום מלמעלה את כמות הפעמים שהאיבר המקסימלי יהיה מוחלף במיון מהיר בעזרת הטענה הבאה:

טענה: מספר ההחלפות המקסימלי של האיבר המקסימלי הוא $n - 1$.
הוכחה: יהא קלט בגודל n. אז יתבצעו עליו p חלוקות כאשר $\lg n \leq p \leq n$.
נשים לב כי במיון מהיר, החלפה יכולה להתבצע אך ורק בין איבר הגדול מאיבר הציר, לאיבר הקטן או שווה לאיבר הציר. לכן, אילו יש בחלוקה כלשהי k איברים הקטנים מאיבר הציר, אז האיבר המקסימלי יכול להתחלף עם לכל היותר $k + 1$ איברים.
בכל חלוקה p_i, נסמן ב k_i את מספר האיברים הקטנים מאיבר הציר. ברור שעבור האיבר האיבר המקסימלי, החלוקה הm תכלול $n - \sum_{i=1}^{m}(k_i + 1)$, שכן "נפטרים" מ$k_i$ האיברים הקטנים מאיבר הציר ומאיבר הציר עצמו בכל חלוקה.
בחלוקה ה p יהיו $n - \sum_{i=1}^{p}(k_i + 1) = 1$ איברים על מנת שהשגרה תפסיק.
לכן, $\sum_{i=1}^{p}(k_i + 1) = n - 1$. במילים אחרות, מספר האיברים הקטנים או שווים לאיברי הציר בכל החלוקות (מספר האיברים המקסימלי איתם היה יכול האיבר המקסימלי להתחלף), הוא $n - 1$ וסיימנו.

נוכיח כי חסם זה הדוק ע"י הצבעה על קלט בו מספר ההחלפות של האיבר המקסימלי הוא בדיוק $n - 1$. דוגמה לקלט כזה - המערך $[n, 1, 2, ..., n - 1]$ כאשר $n - 1$ הוא איבר הציר. האיבר הראשון n יוחלף עם כל $(n - 2)$ האיברים הקטנים מאיבר הציר ולאחר מכן עם איבר הציר ויתבצעו $n - 1$ החלפות.

ב. ההסתברות להשוואה של האיבר המקסימלי והשני המקסימלי היא 1. נראה זאת בעזרת שתי הטענות הבאות:

טענה: קריאה רקורסיבית למיון מהיר בכלל ולמיון מהיר אקראי בפרט עבור תת-מערך כלשהו $A[i, ..., j]$ משמעותה שבתת-מערך זה נמצאים כל האיברים שערכי המיקום שלהם בטווח $[i, j]$, ורק איברים אלה.
הוכחה: הטענה נובעת ישירות מנכונות אלגוריתם המיון המהיר. בקריאה הרקורסיבית הראשונה, כל האיברים נמצאים בתת-המערך עליו מתבצעת הקריאה וערכי המיקום שלהם הם $1..n$ כנדרש.
בכל קריאה אחרת, אילו בתת-המערך היה איבר שערך המיקום שלו קטן מ-i או גדול מj, אז בוודאות בבחירה קודמת של איבר הציר הוא היה מחולק לתת-מערך אחר.

טענה: קריאה רקורסיבית למיון מהיר אקראי על תת-המערך $A[i, ..., n]$ לכל $1 \leq i \leq n - 1$ תגרור השוואה של האיבר המקסימלי והשני המקסימלי.
הוכחה באינדוקציה חזקה:
<u>בסיס האינדוקציה:</u> הקריאה עבור תת-המערך $A[n - 1, n]$, בו לפי הטענה הקודמת יהיו האיבר המקסימלי והשני המקסימלי, תוביל לבחירה אקראית של אחד משני האיברים בתת-המערך להיות איבר הציר. איבר זה יושווה עם כל האיברים בתת-המערך בשגרת ה $Partition$, ובפרט עם האיבר שלא נבחר. לכן בוודאות במקרה זה שני האיברים יושוו.

<u>צעד האינדוקציה:</u> נניח כי לכל $i+1 \le j \le n-1$, קריאה רקורסיבית לתת-המערך $A[j,\dots,n]$ תגרור השוואה של האיבר המקסימלי והשני המקסימלי. נוכיח את הטענה עבור i:
מהטענה הקודמת נובע כי בתת-המערך $A[i,\dots,n]$ נמצאים האיבר המקסימלי והשני המקסימלי. כעת, נבצע בחירה אקראית של איבר הציר:

אילו נבחר האיבר המקסימלי או השני המקסימלי, אז איבר זה יושווה עם כל שאר האיברים בתת-המערך בשגרת ה $Partition$, ובפרט עם האיבר מבין השניים המקסימליים שאינו נבחר. לכן, איברים אלה יושוו.

אילו נבחר איבר אחר, אז בהכרח ערך המיקום שלו קטן מ$n-1$, שכן קיימים במערך לפחות שני איברים הגדולים ממנו. נסמן מיקום זה ב j כאשר $i \le j \le n-2$. לאחר ביצוע ה $Partition$, בתת-המערך $A[j+1,\dots,n]$ יהיו האיברים המקסימלי והשני המקסימלי, כי ערכי המיקום שלהם הם n ו $n-1$. לכן, קריאה רקורסיבית על תת-מערך זה, כאשר $i+1 \le j+1 \le n-1$, תגרור לפי הנחת האינדוקציה השוואה של האיבר המקסימלי והשני המקסימלי.

מהטענה השנייה נובע באופן ישיר כי שני האיברים המקסימלי והשני המקסימלי יושוו במיון מהיר אקראי.

ג. שני האיברים שההסתברות להשוואתם במהלך מיון מהיר אקראי היא הכי נמוכה הם האיבר המינימלי והמקסימלי. נוכיח זאת בעזרת הטענות הבאות:

טענה: האיבר המקסימלי הוא האיבר בעל הסיכוי הקטן ביותר להשוואה עם האיבר המינימלי.
הוכחה: כאמור, במיון מהיר, על מנת ששני איברים יושוו יש צורך בבחירת אחד מהם להיות איבר הציר בקריאה רקורסיבית מסוימת לשגרת ה $Partition$. ההסתברות לבחירה אקראית של אחד מ-2 איברים מתוך n: $\frac{2}{n}$. זאת ההסתברות שהאיבר המינימלי והמקסימלי יושוו. נניח שנבחר איבר ציר אחר, אז בסוף שגרת ה $Partition$ שני איברים אלו יהיו משני צדדיו השונים של איבר הציר ולא יוכלו להיות מושווים בקריאות רקורסיביות נוספות.

אם ניקח איבר אחר, שערך המיקום שלו j כך ש $2 \le j \le n-1$, אז ההסתברות לבחירת איבר הציר להיות אחד משני האיברים המינימלי והאיבר האחר היא $\frac{2}{n}$. לאחר מכן, ההסתברות ששני האיברים יישארו באותו תת-מערך לאחר החלוקה ויוכלו להיות מושווים בחלוקה אחרת, היא שייבחר איבר ציר הגדול משני האיברים. ישנם $(n-1)-j$ איברים כאלה, וההסתברות לבחירת אחד היא $\frac{n-j-1}{n}$ בהתאם. לאחר חלוקה כזאת, ההסתברות ששני איברים אלה יושוו קיימת (אם אחד מהם נבחר להיות איבר הציר) וחסומה מלמטה בהסתברות לבחירה מתוך $n-1$ איברים.

לכן ההסתברות להשוואת האיבר המינימלי עם האיבר ה j חסומה מלמטה ע"י $\frac{2}{n} < \frac{2}{n} + \frac{n-j-1}{n} \cdot \frac{2}{n-1}$ וסיימנו.

ניתן בקלות להכליל טענה זאת- האיבר המקסימלי הוא האיבר בעל הסיכוי הקטן ביותר להשוואה עם האיבר שערך מיקומו i, מבין האיברים שערך מיקומם $i+1..n$, לכל אינדקס i.

ניתן גם, בקלות, להכליל את הטענה ע"י "הזזת" אינדקס המקסימום. מבין כל האיברים שערך מיקומם $i..1$ עבור i כלשהו, האיבר שערך מיקומו i הוא בעל ההסתברות הנמוכה ביותר להיות מושווה עם האיבר המינימלי.

נסמן $P(i,j)$ ההסתברות להחלפת שני האיברים במערך שערכי המיקום שלהם i,j (ללא הגבלת הכלליות, $i<j$). לפי הטענות הכלליות -

$$P(i,j) > P(1,j) > P(1,n)$$

המסקנה - ההסתברות להשוואת האיברים המינימלי והמקסימלי קטנה יותר מכל הסתברות אחרת, וסיימנו.

שאלה 5

נתונות n נקודות (x_i, y_i) כך ש $i = 1, 2, ..., n$, המתקבלות כקלט לשתי האלגוריתמים שעלינו להציע.
ברור כי את המרחק של נקודות אלו מהראשית ניתן למצוא ב $\Theta(1)$. נגדיר:

$$Radius((x_i, y_i)) = \sqrt{x_i^2 + y_i^2}$$

נכונות האלגוריתם נובעת ממשפט פיתגורס - אורך היתר במשולש שקדקודיו $(0, 0), (x_i, 0), (x_i, y_i)$ הוא שורש סכום ריבועי אורכי הניצבים - x_i ו y_i.

כמו כן, ניתן לבדוק ב $\Theta(1)$ האם נקודה נמצאת על מעגל שרדיוסו r כלשהו. מהגדרת המעגל, נקודה הנמצאת על המעגל חייבת להיות במרחק r ממרכז המעגל. ובמקרה שבו מרכז המעגל הוא ראשית הצירים, המרחק בין נקודה נתונה (x_i, y_i) לראשית הוא $Radius((x_i, y_i))$. לכן, על מנת לבדוק אם נקודה כלשהי נמצאת על מעגל ברדיוס נתון, נבדוק $Radius((x_i, y_i)) = r$.

בנוסף על כך, באמצעות וריאציה של שגרת ה $Select$, ניתן למצוא את הנקודה שערך המיקום שלה, לפי הרדיוס, הוא k כלשהו $(1 \leq k \leq n)$, בסיבוכיות זמן ריצה של $O(n)$. במילים אחרות - ניתן למצוא, בזמן לינארי, נקודה שמרחקה מראשית הצירים הוא המרחק ה-k הכי קטן. הוריאציה היא כזאת: בכל מקום שמושווים בו שני איברים ממערך הקלט $A[i], A[j]$, נשווה במקום את $Radius(A[i]),\ Radius(A[j])$ מבלי לפגוע בסיבוכיות זמן הריצה.

נקרא לשגרה זו $Kth\_Smallest\_Radius$. כאמור, סיבוכיות זמן-הריצה של השגרה לא "נפגעת" עקב הוריאציה, ולכן תהיה $O(n)$ במקרה הגרוע.

סעיף א

האלגוריתם שאציע הוא כלהלן, כאשר A הוא מערך של נקודות בגודל n. (נניח שמערך מתחיל מאינדקס 1).
נמצא רדיוס שערך המיקום שלו הוא $\lfloor \frac{n}{2} \rfloor$. נוכיח בהמשך שאם קיים מעגל כנדרש, אז רדיוסו חייב להיות הרדיוס הזה. לאחר מכן, נעבור על המערך ונספור את כמות הנקודות שהרדיוס שלהן זהה לרדיוס זה. אילו קיימות יותר מ $\lfloor \frac{n}{2} \rfloor$ נקודות, אזי קיים מעגל כזה. אחרת - הוא אינו קיים.

$FindCircle(A)$
  $n \leftarrow A.length$
  $\triangleright$ $Retrieve\ radius\ of\ the\ desired\ circle,\ if\ exists$
  $rad \leftarrow Kth\_Smallest\_Radius(A, 1,\ n,\ \lfloor \frac{n}{2} \rfloor)$
  $\triangleright$ $Initialise\ counter\ and\ count\ points\ on\ circle$
  $count \leftarrow 0$
  $for\ i \leftarrow 1\ to\ n$
    $if\ Radius(A[i]) = rad\ then$ $\triangleright$ $If\ point\ is\ on\ the\ circle$
      $count \leftarrow count + 1$
  $\triangleright$ $If\ number\ of\ points\ on\ the\ circle\ is\ greater\ than \frac{n}{2},\ return\ true,\ otherwise\ false$
  $return\ count > \frac{n}{2}$

הוכחת נכונות:

טענה: אילו קיים מעגל עליו נמצאות יותר מ $\frac{n}{2}$ נקודות, אז מתוך כל הרדיוסים של הנקודות במערך, הרדיוס ה-$\lfloor\frac{n}{2}\rfloor$ הכי קטן הוא רדיוס המעגל.

הוכחה: נניח כי קיים מעגל כנדרש. לכן, יש במערך הקלט יותר מ $\frac{n}{2}$ נקודות שהרדיוס שלהן הוא רדיוס מעגל זה (נסמנו ב r). אילו היינו יוצרים מערך ממוין של רדיוסי הנקודות במערך הקלט, אז היו יותר מ $\lceil\frac{n}{2}\rceil$ מופעים רצופים של r, המתחילים מאינדקס k כלשהו.

כאמור, יש לכל הפחות $\lceil\frac{n}{2}\rceil + 1$ מופעים רצופים של r במערך זה, ולכן נדרש שהאינדקס של כל המופעים, ובפרט של התא האחרון, יהיו מוגדרים היטב.
כלומר: $k + \lceil\frac{n}{2}\rceil + 1 - 1 = k + \lceil\frac{n}{2}\rceil \leq n$ ולכן $k \leq n - \lceil\frac{n}{2}\rceil = \lfloor\frac{n}{2}\rfloor$
כמו כן, מאחר ו $k \geq 1$, האינדקס האחרון חסום מלמטה על ידי $\lceil\frac{n}{2}\rceil + 1$.
כלומר: $k + \lceil\frac{n}{2}\rceil \geq 1 + \lceil\frac{n}{2}\rceil \geq \lfloor\frac{n}{2}\rfloor \geq k$ ובפרט $k + \lceil\frac{n}{2}\rceil \geq \lfloor\frac{n}{2}\rfloor \geq k$ לכל k מוגדר.
לכן, הרדיוס שערך המיקום שלו הוא $\lfloor\frac{n}{2}\rfloor$ תמיד יהיה חלק ממקטע המופעים של r, ומכאן שרדיוס המעגל הוא (בין השאר) הרדיוס ה $\lfloor\frac{n}{2}\rfloor$ הכי קטן וסיימנו.

הוכחת הנכונות: מנכונות השגרה $Select$ (ומכאן מנכונות $Kth\_Smallest\_Radius$) נסיק כי rad קיבל את הרדיוס שערך המיקום שלו הוא אכן $\lfloor\frac{n}{2}\rfloor$.
לפי טענת העזר, אם קיים מעגל כנדרש, אז הרדיוס שלו אכן יהיה rad. במילים אחרות, בוודאות לא קיים מעגל כנדרש שהרדיוס שלו אינו rad. נותר רק לבדוק אם קיים מעגל כנדרש שרדיוסו הוא כן rad.

הבדיקה מתבצעת כמפורש - על ידי ספירת כמות הנקודות שהרדיוס שלהם הוא רדיוס המעגל הרצוי rad. אם מספר נקודות אלה קטן או שווה ל $\frac{n}{2}$, אז לא קיים מעגל כנדרש שרדיוסו rad, ומכאן שמעגל כנדרש לא קיים כלל ולכן האלגוריתם יחזיר $false$.
אילו מצאנו מספר נקודות כנדרש, אז אכן קיים מעגל, והוא $x^2 + y^2 = rad^2$, לכן האלגוריתם יחזיר $true$.

סיבוכיות זמן-ריצה

הפעולה	סיבוכיות זמן-ריצה
בחירת הרדיוס המתאים	$O(n)$
ספירת כל הנקודות שמרחקן מראשית הצירים הוא הרדיוס המתאים.	$O(n)$

סך כל סיבוכיות זמן הריצה - $O(n)$.

סעיף ב'

האלגוריתם שאציע הוא כלהלן, כאשר A הוא מערך של נקודות בגודל n. (נניח שמערך מתחיל מאינדקס 1):

ראשית, נמצא את הרדיוס ה $\lfloor \frac{n}{3} \rfloor$ הכי קטן. לכן, יש $\lfloor \frac{n}{3} \rfloor - 1$ רדיוסים הקטנים יותר מרדיוס זה, ולכן יש $\lfloor \frac{n}{3} \rfloor - 1$ נקודות המתאימות להן שהן קרובות יותר לראשית מהנקודה המתאימה לרדיוס ה $\lfloor \frac{n}{3} \rfloor$ הכי קטן.

לכן, נעבור על כל הנקודות בקלט, ואם הן קרובות יותר לראשית (הרדיוס שלהן קטן יותר) או שהרדיוס שלהן זהה לרדיוס ה $\lfloor \frac{n}{3} \rfloor$ (הנקודה ה-$\lfloor \frac{n}{3} \rfloor$ הכי קרובה), נדפיס אותן.

נשים לב שלא הוגדר לנו מה לעשות כאשר יש שתי נקודות (או יותר) שהרדיוס שלהן זהה לרדיוס ה $\lfloor \frac{n}{3} \rfloor$ הכי קטן. במקרה זה, מאחר ולא ניתן לקבוע איזו נקודה יותר קרובה לראשית, האלגוריתם ידפיס את שתי הנקודות.

האלגוריתם:

$ThirdClosestPoints(A)$

$\quad n \leftarrow A.length$

$\quad \triangleright Retrieve \lfloor \frac{n}{3} \rfloor smallest\ radius$

$\quad rad \leftarrow Kth\_Smallest\_Radius(A, 1, n, \lfloor \frac{n}{3} \rfloor)$

$\quad \triangleright Print\ points\ closer\ to\ the\ origin\ (or\ points\ with\ the\ same\ distance)$

$\quad for\ i \leftarrow 1\ to\ n$

$\quad\quad if\ Radius(A[i]) \leq rad\ then \triangleright If\ point\ is\ closer\ or\ equally\ distant$

$\quad\quad\quad print\ A[i]$

הוכחת נכונות

מנכונות $Select$ (ומכאן מנכונות $Kth\_Smallest\_Radius$) הערך שייכנס ל rad יהיה הרדיוס המתאים לנקודה ה $\lfloor \frac{n}{3} \rfloor$ הכי קרובה לראשית. לכן, עבור כל נקודה, אם הרדיוס שלה קטן או שווה ל rad היא נמצאת ב $\lfloor \frac{n}{3} \rfloor$ הנקודות הקרובות לראשית ולכן היא תודפס.

אחרת, לא תבוצע שום הדפסה ובכך סיימנו.

סיבוכיות זמן-ריצה

הפעולה	סיבוכיות זמן-ריצה
בחירת הרדיוס המתאים	$O(n)$
מעבר כל הנקודות והדפסת הנקודות המתאימות	$O(n)$

סך כל סיבוכיות זמן הריצה - $O(n)$.